BCSE103E -
Computer Programming: Java Lab

LAB Assessment – 2

Name: Dhruv Rajeshkumar Shah
Registration No : 21BCE0611
Date : 2nd November 2022

Question 1

Write a class named '**Student'** with the following:

· Data members include **StudentID**, **FName** and **LName** and **Age**.

· Constructor with argument to initialize the data members.

· A method named '**getDetails'** to read the details of the student.

Write a class Named **MarksAnalysis** that inherits **Student** with the following

• Data members named **Total, Average**.

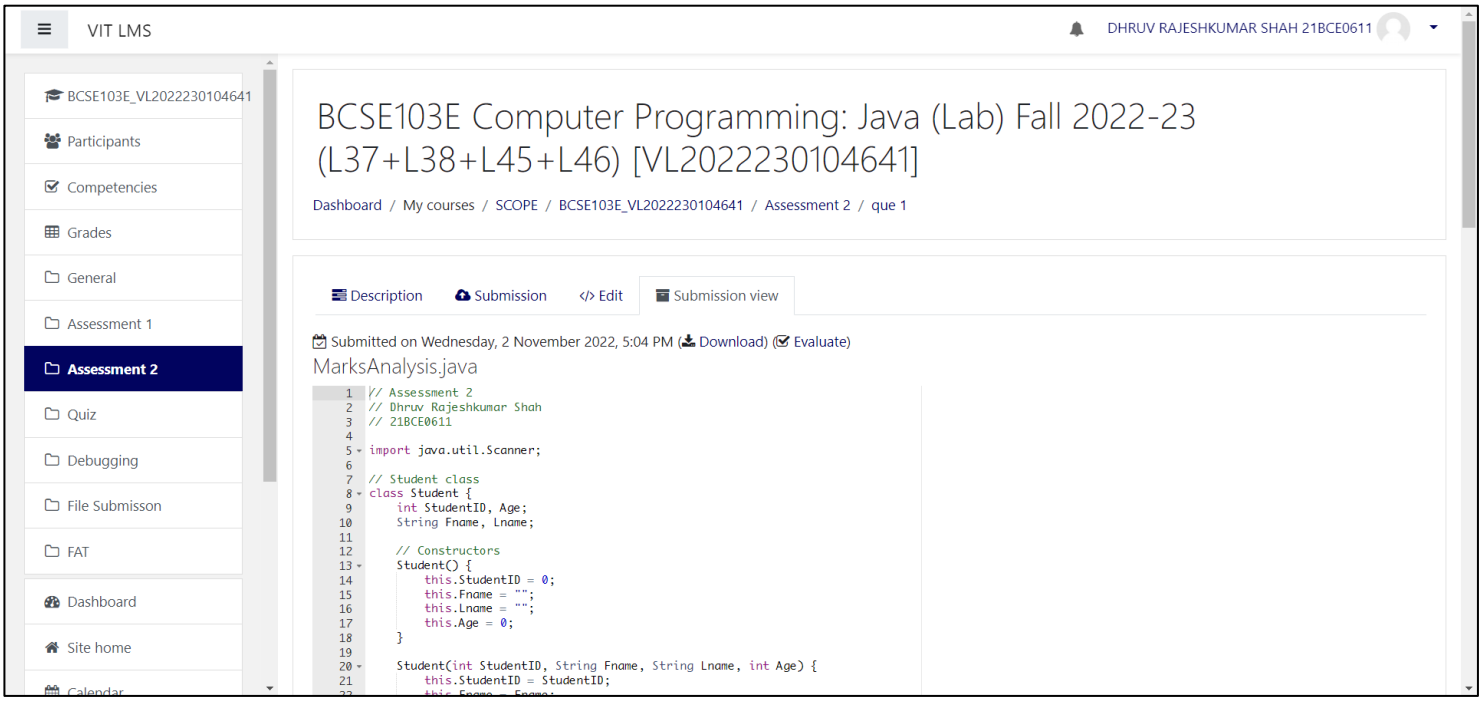
• A method '**findGrade'** will accept the 5 set of marks from the user and find the **Total, Average** and **grade** based on the following table; And print total, average and all the marks and their corresponding grades

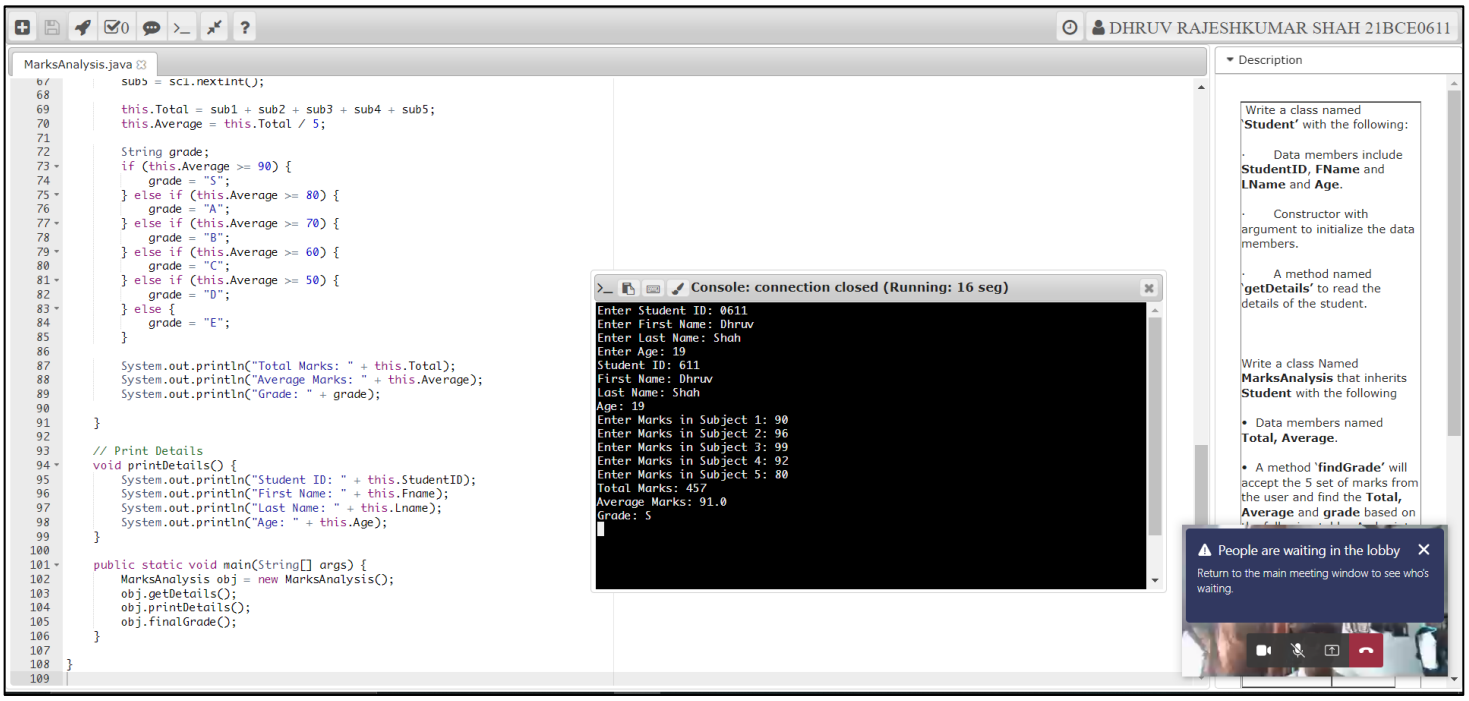
Total	Grade
90 – 100	S
80 – 89	A
70 – 79	B
60 – 69	C
50 – 59	D
<50	E

· A method named '**printDetails'** to display StudentID, FName and LName and Age.

· Start the program execution in the **MarksAnalysis** class and display the student details using the **printDetails** method.

SCREENSHOTS





CODE

```java
// Assessment 2
// Dhruv Rajeshkumar Shah
// 21BCE0611

import java.util.Scanner;

// Student class
class Student {
    int StudentID, Age;
    String Fname, Lname;

    // Constructors
    Student() {
        this.StudentID = 0;
        this.Fname = "";
        this.Lname = "";
        this.Age = 0;
    }

    Student(int StudentID, String Fname, String Lname, int Age) {
        this.StudentID = StudentID;
        this.Fname = Fname;
        this.Lname = Lname;
        this.Age = Age;
    }

    // Get Details
    void getDetails() {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter Student ID: ");
        this.StudentID = sc.nextInt();
        System.out.print("Enter First Name: ");
        this.Fname = sc.next();
        System.out.print("Enter Last Name: ");
        this.Lname = sc.next();
        System.out.print("Enter Age: ");
        this.Age = sc.nextInt();
        // sc.close();
    }

}

public class MarksAnalysis extends Student {
    int Total;
    float Average;

    // Constructor
```

```java
    MarksAnalysis() {
        super();
        this.Total = 0;
        this.Average = 0;
    }

    // Final Grade
    void finalGrade() {
        int sub1, sub2, sub3, sub4, sub5;
        Scanner sc1 = new Scanner(System.in);
        System.out.print("Enter Marks in Subject 1: ");
        sub1 = sc1.nextInt();
        System.out.print("Enter Marks in Subject 2: ");
        sub2 = sc1.nextInt();
        System.out.print("Enter Marks in Subject 3: ");
        sub3 = sc1.nextInt();
        System.out.print("Enter Marks in Subject 4: ");
        sub4 = sc1.nextInt();
        System.out.print("Enter Marks in Subject 5: ");
        sub5 = sc1.nextInt();

        this.Total = sub1 + sub2 + sub3 + sub4 + sub5;
        this.Average = this.Total / 5;

        String grade;
        if (this.Average >= 90) {
            grade = "S";
        } else if (this.Average >= 80) {
            grade = "A";
        } else if (this.Average >= 70) {
            grade = "B";
        } else if (this.Average >= 60) {
            grade = "C";
        } else if (this.Average >= 50) {
            grade = "D";
        } else {
            grade = "E";
        }

        System.out.println("Total Marks: " + this.Total);
        System.out.println("Average Marks: " + this.Average);
        System.out.println("Grade: " + grade);

    }

    // Print Details
    void printDetails() {
        System.out.println("Student ID: " + this.StudentID);
```

```
        System.out.println("First Name: " + this.Fname);
        System.out.println("Last Name: " + this.Lname);
        System.out.println("Age: " + this.Age);
    }

    // Main
    public static void main(String[] args) {
        MarksAnalysis obj = new MarksAnalysis();
        obj.getDetails();
        obj.printDetails();
        obj.finalGrade();
    }

}
```

Question 7

Prime Exception

Create Four user defined exceptions 1. PrimeException 2. NotPrimeException 3.OddException 4.EvenException

Write a program to read numbers from the user and find whether it is prime or not prime and odd or even and throw corresponding exception and handle them with appropriate message to the user.

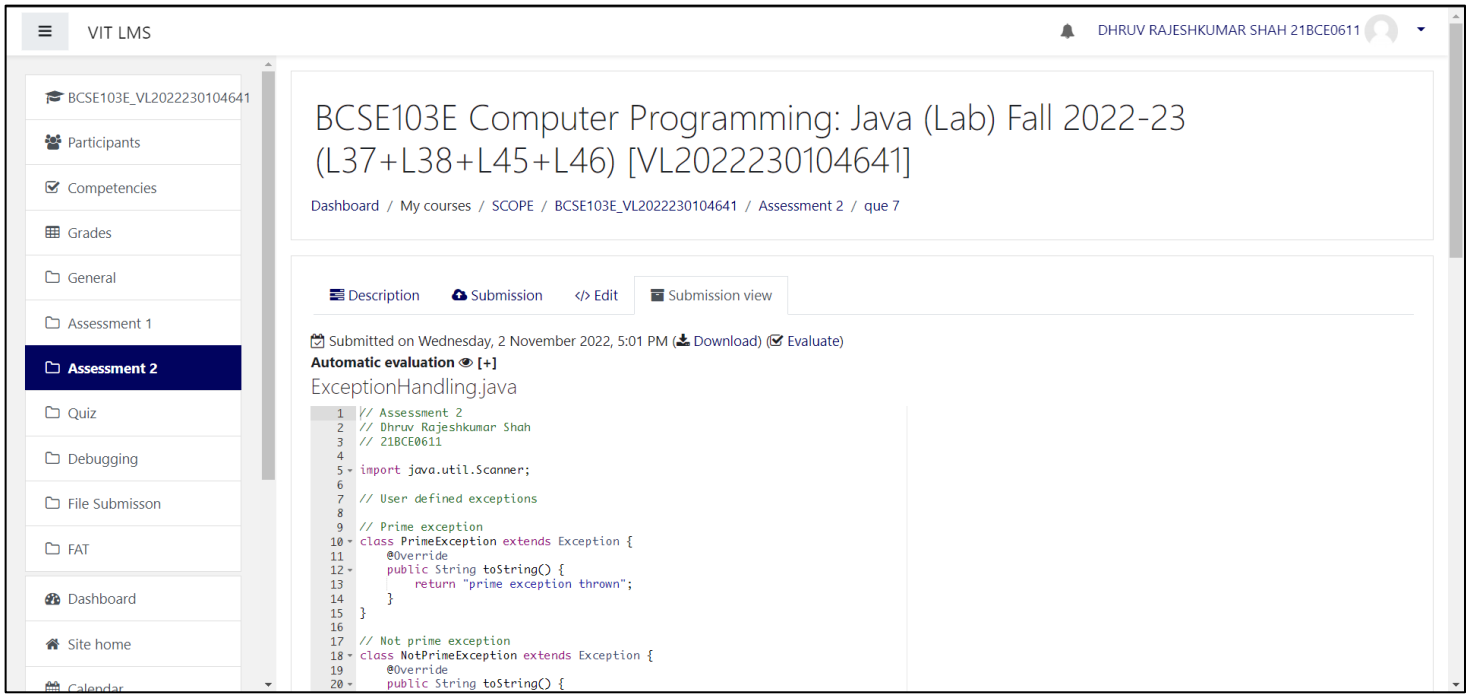
Sample

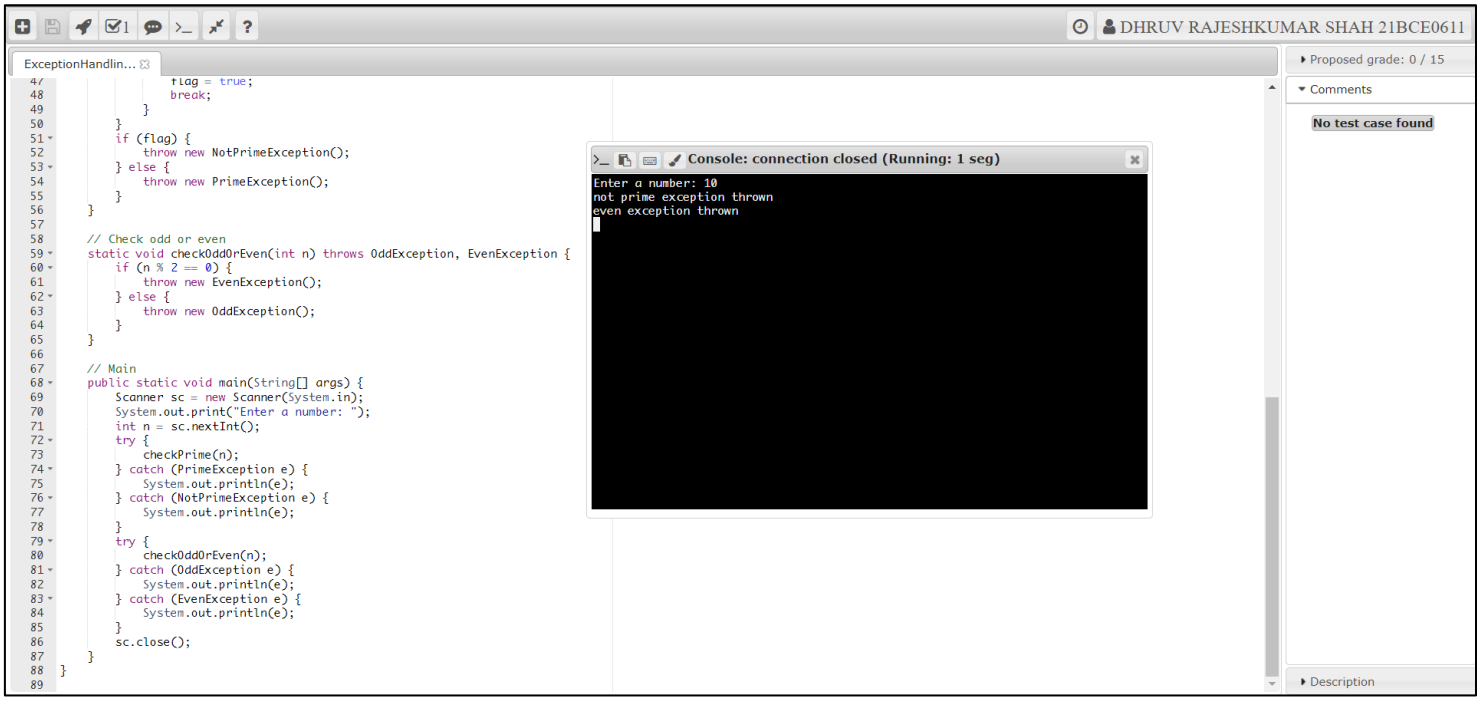
input: 10

output: not prime exception thrown

even exception thrown

SCREENSHOTS





CODE

```
// Assessment 2
// Dhruv Rajeshkumar Shah
// 21BCE0611

import java.util.Scanner;

// User defined exceptions

// Prime exception
class PrimeException extends Exception {
    @Override
    public String toString() {
        return "prime exception thrown";
    }
}

// Not prime exception
class NotPrimeException extends Exception {
    @Override
    public String toString() {
        return "not prime exception thrown";
    }
}

// Odd exception
class OddException extends Exception {
    @Override
    public String toString() {
        return "odd exception thrown";
    }
}

// Even exception
class EvenException extends Exception {
    @Override
    public String toString() {
        return "even exception thrown";
    }
}

public class ExceptionHandling {
    // Check prime or not prime
    static void checkPrime(int n) throws PrimeException, NotPrimeException {
        boolean flag = false;
        for (int i = 2; i <= n / 2; ++i) {
            if (n % i == 0) {
                flag = true;
```

```
                break;
            }
        }
        if (flag) {
            throw new NotPrimeException();
        } else {
            throw new PrimeException();
        }
    }

    // Check odd or even
    static void checkOddOrEven(int n) throws OddException, EvenException {
        if (n % 2 == 0) {
            throw new EvenException();
        } else {
            throw new OddException();
        }
    }

    // Main
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int n = sc.nextInt();
        try {
            checkPrime(n);
        } catch (PrimeException e) {
            System.out.println(e);
        } catch (NotPrimeException e) {
            System.out.println(e);
        }
        try {
            checkOddOrEven(n);
        } catch (OddException e) {
            System.out.println(e);
        } catch (EvenException e) {
            System.out.println(e);
        }
        sc.close();
    }
}
```